(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
16 August 2001 (16.08.2001)

PCT

(10) International Publication Number
WO 01/59690 A1

(51) International Patent Classification[7]: G06K 9/00

(21) International Application Number: PCT/US01/03275

(22) International Filing Date: 31 January 2001 (31.01.2001)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
09/501,355 9 February 2000 (09.02.2000) US

(71) Applicant: VERIDICOM, INC. [US/US]; 2040 Martin Avenue, Santa Clara, CA 95050 (US).

(72) Inventor: RUSSO, Anthony, P.; Apartment #3A, 58 W. 57th Street, New York, NY 10023 (US).

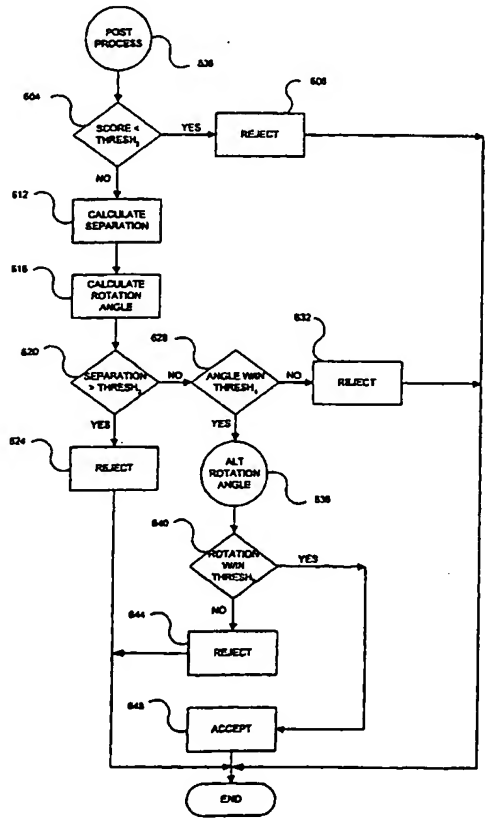
(74) Agent: WOLFF, Jason, W.; Lyon & Lyon LLP, Suite 4700, 633 West Fifth Street, Los Angeles, CA 90071-2066 (US).

(81) Designated States *(national)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CR, CU, CZ, DE, DK, DM, DZ, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, UZ, VN, YU, ZA, ZW.

(84) Designated States *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

[Continued on next page]

(54) Title: BIOMETRIC FALSE ACCEPT DETECTION



(57) Abstract: Techniques and apparatuses for detecting false accepts in biometric authentication systems are provided. According to an aspect of the invention, a sequence of threshold tests are performed on measured biological data after it is compared to valid stored data. According to one embodiment, the threshold tests comprise a distance test (612), a rotational angle test (616), and, optionally, an alternate rotation angle test (636). In a fingerprint sensing environment, the distance test (612) examines the separation between selected minutiae in a measured fingerprint and corresponding minutiae in a stored constellation (300). In the rotational angle test (616), a total angle difference between selected minutiae in the measured fingerprint and corresponding minutiae in the stored constellation (300) is examined. In the alternate rotation angle test (636), a Gestalt rotational difference between selected minutiae in the measured fingerprint and in the stored constellation (300) is calculated. Thus, two independent calculations of the rotation angle between the sensed and valid constellations can be reconciled. In each of the tests, threshold parameters provide a cutoff for rejecting one or more matched minutia that are actually "false accepts".

WO 01/59690 A1

Published:
— *with international search report*
— *with amended claims and statement*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

TITLE OF THE INVENTION:

BIOMETRIC FALSE ACCEPT DETECTION

BACKGROUND OF THE INVENTION

1. Field of the Invention

This invention is related to the field of biometrics, and in particular to detecting errors when validating biometric data.

2. Background

Biometric data is frequently used to verify or authenticate a particular individual who desires to gain access to some facility, be it a door entry system, a computer system, or a particular level of authorization in a computer program.

A drawback to many computer systems is that they employ passwords, which must be memorized. Users frequently forget their passwords or they write them in non-secure place. Biometric sensors, which measure certain biological properties and use statistical analysis to validate a user, generally provide a better solution to passwords, at least in so far as the biometric data used to validate the user is always present -- yet unknown. This is to say that there is no password that needs to be memorized and there is less risk of the validating information being compromised.

One drawback to biometric systems is that wide-spread deployment is generally not practicable. At least one reason for this is that biometric systems often employ an

"escrow" functionality. The escrow functionality generally involves storage of a valid set of biometric data in a remote, presumably secure, server (hereinafter an "escrow server"). A measuring function typically records a user's biometric data and then compares the recorded data with data in the escrow server. If the data matches, then the escrow server may issue a response that the user passed a biometric test.

However, having a central escrow server creates problems. First, a cracker may break into the escrow server and steal or manipulate the stored biometric data. Thus, privacy becomes an issue. Second, an unauthorized server may issue false indicators that particular recorded data matched the stored value in the escrow server.

A solution to the escrow server is to employ a smartcard based system. A smartcard is small, portable card (much like a credit card) consisting of a memory that persistently stores the biometric data. When access to a particular facility is desired, a user simply inserts the smartcard into a smartcard reader and the reader extracts the biometric data from the smartcard. The reader subsequently compares the extracted biometric data against measured biometric data from the biometric sensor. Alternatively, the biometric sensor provides the measured biometric data to the smartcard for processing.

However, even this approach has drawbacks. One drawback is that smartcards, because of their physical constraints, are usually low-power systems that do not allow for a sophisticated microprocessor. Instead, a small instruction set microprocessor is used that has limited computational abilities and a limited memory. Due to the limited computational power of a smartcard, tradeoffs in accuracy need to be made if biometric data is to be verified in a reasonable amount of time. Therefore, it is possible that when the smartcard microprocessor attempts to compare measured data from a biometric sensor and a template stored in the smartcard, a certain number of "false accepts" occur. That is, the smartcard may erroneously accept biometric data that does not truly match the stored template, or the smartcard may accept biometric data that is a spoof of the item from which biometric data is collected.

SUMMARY OF THE INVENTION

A method and apparatus for detecting false accepts in a biometric authentication system is provided. According to an aspect of the invention, a sequence of threshold tests are performed on measured biological data after it is compared to valid stored data.

According to one embodiment, the threshold tests comprise a distance test, a rotational angle test, and, optionally, an alternate rotation angle test.

In a fingerprint sensing environment, the distance test examines the separation between selected minutiae in a measured fingerprint and corresponding minutiae in a stored constellation. In the rotational angle test, a total angle difference between selected minutiae in the measured fingerprint and corresponding minutiae in the stored constellation is examined. In the alternate rotation angle test, a Gestalt rotational difference between selected minutiae in the measured fingerprint and in the stored constellation is determined. Thus, two independent calculations of the rotation angle between the sensed and valid constellations can be reconciled.

In each of the tests, threshold parameters provide a cutoff for accepting or rejecting a measured fingerprint. According to one embodiment, the tests are sequentially performed and if any test fails, then the matched minutiae are rejected as "false accepts".

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 depicts a smartcard system comprising a smartcard, a smartcard reader, and a biometric sensor.

FIG. 2A depicts fingerprint and select minutia.

FIG. 2B depicts data parameters of a selected pair of minutiae.

FIG. 3 depicts a data structure in a fingerprint template.

FIG. 4 depicts data structures for matched fingerprint minutiae.

FIG. 5 is a flowchart depicting an improved fingerprint template matching technique.

FIGS. 6-7 are flowcharts depicting false accept processing.

FIGS. 8A-C are conceptual schematics of a minutiae distance test.

FIGS. 9A-C are conceptual schematics of a minutiae angle test.

FIGS. 10A-C are conceptual schematics of a minutiae rotation angle test.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

FIG. 1 depicts a smartcard system comprising a smartcard 100, a smartcard reader 110, and a biometric data sensor 116. The smartcard 100 typically comprises a microprocessor 104 and a RAM 106, EEPROM 103(or equivalent persistent yet

programmable memory, such as "flash") and ROM 102, each communicatively coupled to the microprocessor 104. Executable software code (also called a "computer program") resides in ROM 102. RAM 106 is used for storage of variables and other runtime data need by the microprocessor 104. The smartcard reader 110 includes a microprocessor 114 and RAM 112. The smartcard reader 110 may further comprise a ROM, in which executable software code resides, or the software code may be loaded over a bus 124 from an external storage device or computer. In operation, the smartcard 100 is inserted into the smartcard reader 110, wherein data is either transferred from the smartcard to the smartcard reader, or from the smartcard reader 110 to the smartcard 100

The smartcard system depicted in FIG. 1 further comprises a biometric sensor 116, which is preferably a fingerprint sensor. The sensor 116 is configured to receive a finger. When a finger is placed over the sensor 116, the sensor 116 scans the finger's print and creates a fingerprint image. (It is worth noting that the image can be an analog signal that is later converted into a digital representation or "template".) A presently preferred fingerprint sensor is the FPS110 available from Veridicom Corporation in Santa Clara, California <http://www.veridicom.com>, or alternatively the Veridicom 5th Sense (TM) peripheral sensor.

An exemplary sensed fingerprint image 200 is shown in FIG. 2A. A series of ridges have endpoints and bifurcations, which are commonly described as "minutia" 204. Select minutia from a sensed fingerprint are converted from their sensed form to a digital form as a data structure, one embodiment being a multidimensional fingerprint constellation (or simply "constellation"). An embodiment of a constellation is described below with reference to FIG. 3.

When stored as a constellation, parameters are chosen to define the relationship between individual minutiae 204 in the fingerprint. Referring to FIG. 2B, a geometric representation 250 of two minutiae is shown. A first minutia 254 is selected as a reference minutia. A tangent 264 to the ridge of the first minutia 254 is determined and then extrapolated to intersect an x-axis 272 (for example, an x-axis of the sensor 116). An angle "alpha" is measured between the x-axis 272 and the tangent line 264. The x- and y-coordinates of the minutia 254 are also recorded. The first minutia 254 is then used a reference point to compare relative values with neighboring minutiae.

For example, a second minutia 258 is selected as a neighbor to minutia 254. X- and y-coordinates of the second minutia 258 are recorded, thus a distance 262 between the first minutia 254 and the second minutia 258 can be determined, as can an angle "beta" -- formed between lines 272 and 268. The difference between angle alpha and angle beta, or angle "gamma" can be calculated by subtracting one angle from the other. Angle gamma can be represented as the angle formed between tangent line 264 (minutia 254) and tangent line 268 (minutia 258).

The process described above can be repeated comparing a third, fourth, fifth, etc. minutia to the first minutia 254. Finally, a multidimensional fingerprint constellation can be created that represents the relative values of the selected minutiae in a particular neighborhood. FIG. 3 depicts such a data structure.

The multidimensional constellation 300 comprises a header 304 that can be used to identify the make and version of the multidimensional constellation 300. A reference identifier 312 gives the x-, y- and alpha-values for the reference (or first) minutia that is the basis for relative values in a neighborhood constellation 308. The neighborhood constellation 308 further comprises a plurality of data values for the neighbors of the reference identifier 312 (that is, the second, third, fourth, etc. minutia). There can be as many there can be as many neighborhoods 308 in a multidimensional fingerprint constellation 300 as are desired. According to one embodiment, there are one-hundred neighborhood constellations 308 in a multidimensional constellation 300.

Three columns represent the relative values between the minutiae in each neighborhood constellation 308. They are columns 320, 324, and 328, which represent a relative distance, an angle, and a relative angle difference, respectively, as is discussed above with reference to FIG. 2B. There are as many neighbors 316 in a neighborhood constellation 308 as are desired. According to one embodiment, there are fifteen neighbors 316 in a neighborhood constellation 308. (It is noted that when data is stored in the multidimensional constellation 300, it can be normalized so that the values have a uniform bit-size.)

Returning to FIG. 1, as used with the smartcard system, the fingerprint image measured by sensor 116 is transferred to RAM 112 as a multidimensional fingerprint constellation 300. For purposes of explanation, the sensed multidimensional fingerprint

constellation will be referred to hereinafter as "template B", while a valid multidimensional fingerprint constellation will be referred to hereinafter as "template A".

When the smartcard 100 is inserted into the smartcard reader 110, information is exchanged over a communication channel 120, be it physical or wireless, between the smartcard 100 and the smartcard reader 110. According to an embodiment of the invention, sensed template B is serially transferred, one neighborhood (or subset of template B) at a time, to RAM 106, where it can be compared by microprocessor 104 to the valid template A that is stored in the smartcard 100, for example in EEPROM 103.

FIG. 5 is a flowchart depicting generalized steps for matching fingerprint templates. Specific details of an embodiment of the generalized steps are described in co-pending U.S. Patent Application Serial No. 09/354,929.

Referring to FIGS. 1 and 5, in step 504 biometric data, for example a fingerprint, is read from sensor 116. A fingerprint template (template B) is created and placed in RAM 112 in step 508. The first two steps are performed by the smartcard reader 110 and sensor 116, while the subsequent steps are generally under control of the microprocessor 104.

In step 512, a subset of template B is read from RAM 112 and transferred to RAM 106. The subset is preferably a neighborhood constellation 308. In step 520, the subset of data from template B is compared to sequential valid neighborhood constellations in template A. If the template data matches, then step 520 passes processing on to step 528. Otherwise, step 520 passes processing on to step 524, where the microprocessor 104 causes a next subset of template B to be transferred from RAM 112 to RAM 106.

In step 528, template match sets are created for each reference value corresponding to the matched minutiae in templates A and B. If the reference values corresponding to FIG. 2B are used, then three template match sets are created -- one for the x-, y-, and beta-values.

FIG. 4 depicts an exemplary template match set 400. A first column 404 holds the matched x-, y- or beta-values for template A, and a second column 408 holds the corresponding x-, y- or beta-values for template B. There are as many rows 412 in the match sets as there are matched minutia (or "matches") between templates A and B. According to one embodiment, each template match set 400 holds four matched pairs. Each match corresponds to a "score". The more matches, the higher the score.

In step 532, the score for the matches is compared against a first threshold value. If the score is greater than the first threshold value, then the matches are determined to be valid and not "false accepts". Accordingly, processing continues to step 540 where the minutia match is accepted. However, if the score is less than or equal to the first threshold value, then post processing occurs at step 536. According to one embodiment, the first threshold value is the value four. If greater than four matches are found, then the odds of a false accept are low. However, if between one and four minutia matches are found, then the odds of a false accept are higher, thus a finer granularity of testing for a match is desired.

FIG. 6 is a flowchart depicting post processing of template matches that have been marked as potential false accepts. In step 604, the score is compared against a second threshold value. If the score is less than the second threshold value, then the matches are rejected -- as no meaningful finer granularity testing is possible. For example, the second threshold value can be three, so if one or two matches are found, then they are rejected in step 608 as false accepts because more certainty in the matches is desired.

In step 612, a total separation between the minutiae in template A and then the minutiae in template B is calculated. FIGS. 8A-C show conceptual diagrams of an embodiment of a minutiae distance test corresponding to step 612. FIG. 8A shows the total separation between a series of minutiae in template A, while FIG. 8B shows the same in template B. FIG. 8C shows how the two total separations are compared to a third threshold value.

According to one embodiment, the total separation is determined by a total distance difference. In the total distance difference test, the total amount of separation between the minutiae in each template is calculated. For example, the x-value for the second match of template A is subtracted from the x-value of the first match of template A. Next, that value is squared. Then the y-value for the second match of template A is subtracted from the y-value of the first match of template A. This result is also squared. Finally, a sum of the squares of the total x- and y- differences is calculated, which yields a total distance difference or "separation" value. An equation for the total distance difference is:

$$\sum_{i=1}^{i=score}[(x_{i-1}-x_i)^2+(y_{i-1}-y_i)^2] \quad (1)$$

As indicated in Eq. (1), the process is repeated for each x- and y-value stored in the template match sets corresponding to template A. It is also repeated for the x- and y-values corresponding template B. In an alternative embodiment, rather than squaring the difference, the sum of the absolute values of the differences can be compared, which can be less of a drain on the microprocessor's computational resources.

In step 616 a rotation angle is calculated. Conceptual diagrams of an embodiment of a minutiae angle test corresponding to step 616 are shown in FIGS. 9A-C. FIGS. 9A and 9B show the beta angles between each minutia in templates A and B, respectively. FIG. 9C, while showing only a single rotation angle, represents the broader concept. The sum of the differences of the beta angles (between templates A and B) is actually what is compared to the fourth threshold value. An equation for the total rotation angle, which represents the amount of rotation between the sensed and valid templates, is:

$$\frac{\sum_{i=0}^{i=score-1}[\beta_{A,i} - \beta_{B,i}]}{score} \quad (2)$$

In step 620 the separation calculated in step 612 is compared against the third threshold value. The third threshold amount can be any amount chosen by one who practices the invention and can vary depending on the types of minutia selected. For example, a greater tolerance may be expected when the minutiae are selected so as to maximize their separation, or when many minutiae are selected. However, a lesser amount of tolerance may be desired when the minutiae are selected base on their close proximity to the reference minutia. If the separation value is greater than the third threshold value, then the match is rejected as a false accept in step 624. According to an alternative embodiment, if the number of matches to be compared falls within a small range, dividing the sum by the score is not necessary because accommodation for the overestimate of the rotation angle can be accounted for in the selection of the fourth threshold value.

However, if the separation value is not greater than the third threshold value, then in step 628 the angle error from step 616 is compared to the fourth threshold value. If the angle error is outside of the fourth threshold value, then the match is rejected as a false accept in step 632. Again, the fourth threshold value can be chosen arbitrarily by one who implements the invention described herein. As was the case in step 620, the smaller the

threshold value, the lower the odds that a false accept will occur and consequently greater certainty in the actual template matchings.

However, if the rotation angle is within the fourth threshold value, then in step 636 an alternative rotation angle between the two templates (or neighborhoods) is calculated. FIGS. 10A-C are conceptual diagrams of what is achieved by the alternate rotation angle calculation. FIG. 10A is a diagram of minutiae in template A. FIG. 10B is a diagram of minutiae in template B. FIG. 10C is a diagram showing the rotation angle between the minutiae of template A and template B, as well as a fifth threshold value. FIG. 7, described in detail below, shows one embodiment for calculating the alternate rotation angle in step 636.

In step 640, the alternate rotation angle is compared to the fifth threshold value. If the alternate rotation angle is within the fifth threshold value, then the template matches are accepted in step 648. However, if the alternate rotation angle is not within the fifth threshold value, then it is rejected in step 644. If the accept in step 648 occurs, then the template matchings are determined to be valid and not false accepts.

FIG. 7 depicts an embodiment of alternate rotation angle calculation. According to one embodiment, the alternate rotation angle is calculated so that two independent measures of the rotation between templates A and B can be compared. The first method uses the difference in the angles of the matched minutia pairs averaged over all the pairs (step 616). The second method, shown in FIG. 7 and 10A-C, uses a geometrical construct that calculates the centroids of the set of matched minutiae in each template and yields a Gestalt rotational difference.

In step 704, a centroid of template A is calculated. This is done by separately summing the x- and then the y-values for the template A values in the match templates 400, then dividing the totals by the total number of matches (or score). This normalizes the centroid for the neighborhoods of templates A and B. The same process is repeated for the x- and y-values for the template B values in the match templates 400 in step 708.

Eq. (3) shows the result obtained by steps 712 through 776. In particular, steps 712 through 748 represent the numerator of Eq. (3), while steps 752 through 772 represent the denominator of Eq. (3). Step 776 shows the division of the numerator by the denominator, while step 780 shows one embodiment of how the two independent calculations of the rotation angle can be compared.

$$\frac{\sum_{i=0}^{i=score}[(x_{A,i}-centroid_{A,x})*(x_{B,i}-centroid_{B,x})+(y_{A,i}-centroid_{A,y})*(y_{B,i}-centroid_{B,y})]}{\sum_{i=0}^{i=score}[(y_{A,i}-centroid_{A,y})*(x_{B,i}-centroid_{B,x})-(x_{A,i}-centroid_{A,x})*(y_{B,i}-centroid_{B,y})]} \quad (3)$$

The result of Eq. 3 is an approximation of the tangent of the rotation angle (in radians). When a greater degree of precision is needed so as to avoid approximation errors, successive approximation techniques or amplification methods can be employed when the results yield an angle approaching zero and +/- ninety degrees. Moreover, since most microprocessors used in a smartcard will not necessarily have the ability to perform floating point arithmetic, it is preferred that when the numerator or denominator is zero, that an appropriate rotation for the two templates be assumed (for example zero or ninety degrees). While most microprocessors in a smartcard 100 will have minimal mathematical functionality, the approximation made by Eq. (3) can obviously be replaced if geometric functions (e.g., sine, cosine, tangent) are included with the microprocessor's instruction set.

After step 776, the process continues to step 780, where the first rotation angle(616) is subtracted from the second (776). In step 784, the flow returns to step 640 (FIG. 6). It is noted that step 780 is not necessary, although it is used in a presently preferred embodiment as a way to reconcile the first and second independent rotation angle calculations -- ideally, the difference between the two rotations is zero.

It is noted that the test in step 640 can be changed depending on whether step 780 is performed. If step 780 is not performed, then the rotation angle determined by Eq. (3) can be compared the fourth threshold value. According to one embodiment, a rotation angle outside of 50 degrees is considered unacceptable and the match is rejected as a false accept. Otherwise, the test succeeds and the matches are accepted.

The steps for biometric false accept error processing are preferably embodied in an executable computer program that is stored in a computer readable medium, such as the ROM 102, as one or more sequences of instructions that impart technical functionality into a computer processing system. Just prior to execution, the one or more sequences of instructions are loaded in to an execution memory area of the microprocessor 104 from which they are ultimately performed. Not all of the steps have to be performed by a single microprocessor, such as processor 104. Rather, some of the instructions can be stored separate from the smartcard 100 and can be executed by a processing unit completely

apart from the smartcard 100, for example in microprocessor 114, or in another remote unit coupled to either the smartcard 100 or the smartcard reader 110. Moreover, selected steps can be performed by executable object code, while others are interpreted and then executed at runtime.

According to one embodiment, a personal computer will be coupled to the smartcard reader and the fingerprint sensor. Moreover, the personal computer will have connectivity to the Internet or an intranet, for example through an internet access provider, an internet service provider, or a local area network. A remote server on the network, the remote server including an application program such as a HTTP, CGI, ASP, Perl or XLM based banking or information system, will request authentication of a user at the personal computer. In turn, the personal computer will initiate instructions that cause the biometric sensor to sense biometric data from the user, and then cause the smartcard reader to validate the sensed data with data stored on the smartcard inserted into the smartcard reader. Once the smartcard has validated the sensed biometric data, an accept or reject signal is then transferred back to the personal computer. The personal computer can, in turn, send the accept, reject, or another acknowledgement signal back the remote server. Networking protocols such as TCP/IP or ATM, which are generally known in the art, can provide the connectivity transport between the computer systems on the network.

Upon review the detailed description and drawings, one of skill in the art could make certain modifications or improvements to the subject invention, yet not depart from its overall spirit. For example, cryptographic functions may be used to encode data on the smartcard 100, or to protect communication channels from rote copying. For example, the techniques described in U.S. Patent Application Serial No. 09/306,148, filed April 26, 1999, entitled, "HIGH SECURITY BIOMETRIC AUTHENTICATION OF USE OF PUBLIC KEY IN A PUBLIC KEY ENCRYPTION SYSTEM" can be employed. In another embodiment, the actual initial tests performed on the templates can include tests to detect spoofing. For example, the methods disclosed in U.S. Provisional Application 60/158,423, filed October 7, 1999, entitled "METHOD AND APPARATUS FOR DETERMINING A LIVING FINGER ON A FINGERPRINT SENSOR" can be employed.

Further, the false accept error processing techniques can be moved completely off of the smartcard 100 and into another device, such as a computer or server attached to the smartcard reader 110 or the sensor 116.

Numerous other modifications can be made, such as different ways of comparing a measured value to a valid value, or matched values to threshold values. Other normalization techniques can also be employed, such as conversion of angles to particular quadrants, or taking absolute values of the matched values. Even the data structures could be modified so they are modestly different or are amalgamations of the structures described above. Furthermore, the comparisons to threshold values can be made immediately after individual or particular calculations are made, or after all the calculations are made, so that a minimal number of instruction cycles are used when false accept error processing. Accordingly, the invention is to be interpreted consistent with the accompanying claims, rather than strictly limiting it to the detailed embodiments described above.

CLAIMS

What is claimed is:

1. A method for detecting a false accept of a biometric template match, comprising:

calculating (612) a separation distance between a plurality of sensed minutiae and a plurality of valid minutiae;

comparing (620) said separation distance with a threshold separation distance;

rejecting (624) said biometric template match separation distance is outside of said threshold separation distance;

calculating (616) a rotation angle between said plurality of sensed minutia and said plurality of valid minutiae;

comparing (628) said rotation angle with a threshold rotation angle; and

rejecting (632) said biometric template match if said rotation angle is outside of said threshold rotation angle.

2. The method of claim 1, further comprising:

calculating (636) an alternate rotation angle between said plurality of sensed minutiae and said plurality of valid minutia;

comparing (640) said alternate rotation angle to a threshold alternate rotation angle; and

rejecting (644) said biometric template match if said alternate rotation angle is outside of said threshold alternate rotation angle.

3. The method of claim 2, wherein said act of calculating said alternate rotation angle comprises:

determining (704) a first centroid of said plurality of valid minutiae;

determining (708) a second centroid of said plurality of sensed minutiae;

determining (720) a first difference between a first coordinate of one of said plurality of valid minutia and a first coordinate of said first centroid;

determining (724) a second difference between a first coordinate of one of said plurality of sensed minutia and a first coordinate of said second centroid;

determining (732) a third difference between a second coordinate of one of said plurality of valid minutia and a second coordinate of said first centroid;

determining (736) a fourth difference between a second coordinate of one of said plurality of sensed minutia and a second coordinate of said second centroid;

multiplying (728) said first difference by said second difference to create a first product;

multiplying (740) said third difference by said fourth difference to create a second product; and

summing (744) said first product and said second product to create a first portion of said alternate rotation angle.

4. The method of claim 3, further comprising:

multiplying (760) said third difference by said second difference to create a third product;

multiplying (764) said first difference by said fourth difference to create a fourth product;

subtracting (768) said fourth product from said third product to create a second portion of said alternate rotation angle; and

dividing (776) said first portion of said alternate rotation angle by said second portion of said alternate rotation angle to create a value corresponding to said alternate rotation angle.

5. The method of claim 2, wherein said act of comparing said alternate rotation angle includes:

calculating (780) a difference between said alternate rotation angle and said rotation angle; and

comparing said difference between said alternate rotation angle and said rotation angle to said threshold alternate rotation angle.

6. The method of claim 2, wherein said act of calculating said rotation angle comprises averaging a difference between a plurality of sensed minutia angles and a plurality of matched minutia angles.

7. The method of claim 2, wherein said step of comparing said alternate rotation

angle to said threshold alternate rotation angle includes subtracting said alternate rotation angle from said rotation angle.

8. A computer program for execution in one or more biometric authentication devices, the computer program configured to cause one or more microprocessors to perform the steps in any of claims 1 through 7.

9. A smartcard comprising:

a microprocessor (104);

a random access memory (106) communicatively coupled to said microprocessor (104), said random access memory (106) configured to hold at least a portion of a first biometric template that corresponds to a sensed object;

a persistent memory (103) communicatively coupled to said microprocessor (104), said persistent memory (103) configured to hold a second biometric template that corresponds to a valid object; and

a read only memory (102) communicatively coupled to said microprocessor (104), said read only memory (102) including executable object code configured to cause said microprocessor (104) to:

create one or more match templates (400) representing similar minutiae (412) between said first biometric template and said second biometric template;

compare said one or more match templates (400) using a distance test (612), a first rotation angle test (616), and a second rotation angle test (636); and

accept one or more match templates (400) if all of said tests are successful.

10. The smartcard of claim 9, wherein said executable object code is further configured to cause said microprocessor (104) to reconcile said first rotation angle test (616) with said second rotation angle test (636).

11. The smartcard of claim 9 or 10, wherein said distance test (612) is defined by the formula

$$\sum_{i=1}^{i=score}[(x_{i-1}-x_i)^2+(y_{i-1}-y_i)^2],$$

said first rotation test (616) and said second rotation angle tests (636) are, alternatively, defined by the formulas

$$\frac{\sum_{i=0}^{i=score-1}[\beta_{A,i}-\beta_{B,i}]}{score} \text{ and}$$

$$\frac{\sum_{i=0}^{i=score}[(x_{A,i}-centroid_{A,x})*(x_{B,i}-centroid_{B,x})+(y_{A,i}-centroid_{A,y})*(y_{B,i}-centroid_{B,y})]}{\sum_{i=0}^{i=score}[(y_{A,i}-centroid_{A,y})*(x_{B,i}-centroid_{B,x})-(x_{A,i}-centroid_{A,x})*(y_{B,i}-centroid_{B,y})]}.$$

12. A biometric authentication system comprising:

a smartcard (100) comprising:

a microprocessor (104),

a random access memory (106) coupled to said microprocessor (104) and configured to hold a set of sensed biometric information,

a persistent memory (103) coupled to said microprocessor (104) and configured to hold a set of valid biometric information;

a read only memory (102) coupled to said microprocessor (104) and configured to hold executable biometric testing software, and

a first communication means communicatively coupled to said microprocessor;

a smartcard reader (110) comprising second communication means, said second communication means configured to exchange information with said first communication means and second communication means; and

a biometric sensing means (116) communicatively coupled to said smartcard (100) through second communication means, said biometric sensing means (116) configured to collect sensed biometric information from an object and allow said smartcard reader (110) to send said sensed biometric information to said smartcard (100);

wherein said microprocessor (104) in said smartcard (100), through said executable biometric testing software, is configured to compare said set of sensed biometric information to said set of valid biometric information, create a set

of matched biometric information (400), and verify said set of matched biometric information (400) using a distance test (612) and a first rotation angle test (616).

13. The biometric authentication system of claim 12, wherein said microprocessor (104) in said smartcard (100) is further configured to verify said set of matched biometric information (400) using a second, independent, rotation angle test (636).

14. The biometric authentication system of claim 12 or 13, wherein said set of sensed biometric information and said set of valid biometric information are stored a data structure representing a multidimensional fingerprint constellation (300).

15. The biometric authentication system of claim 12, 13 or 14, further comprising a first computer coupled to said smartcard reader (110) and said biometric sensor (116), said first computer configured to initiate and dispatch data and commands to and from said smartcard (100) and said biometric sensor (116).

16. The biometric authentication system of claim 15, further comprising:

a second computer comprising a networked application, said second computer communicatively coupled to said first computer through a communications network, said second computer configured to send and receive additional data and commands to and from said first computer from said networked application, said additional data and commands including information that causes said biometric authentication system to activate; and

said first computer further configured to communicate over said communications network to said second computer.

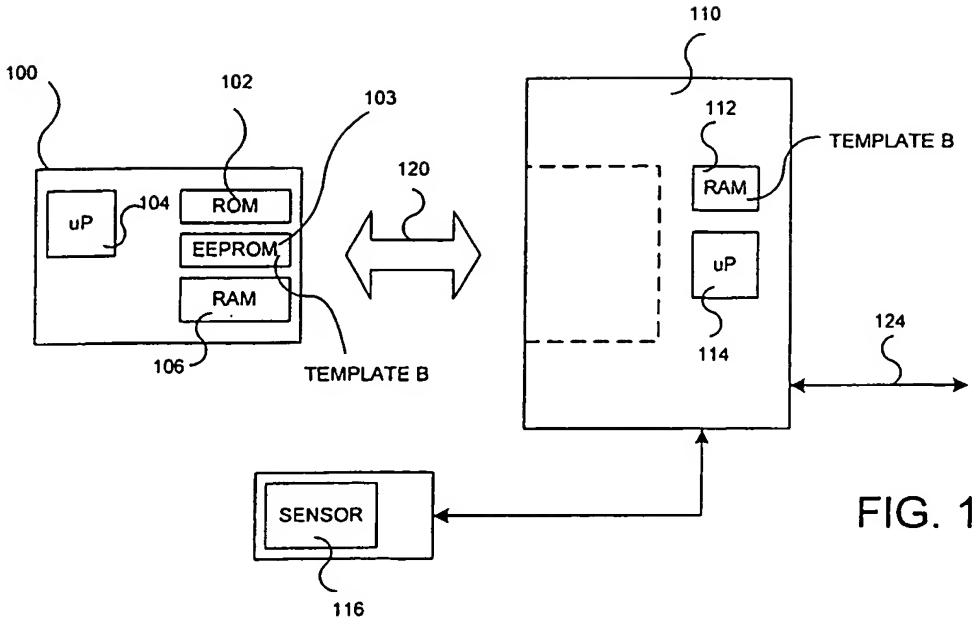

100
102
103
104
uP
ROM
EEPROM
RAM
106
TEMPLATE B
120
110
112
TEMPLATE B
RAM
uP
114
124
SENSOR
116

FIG. 1

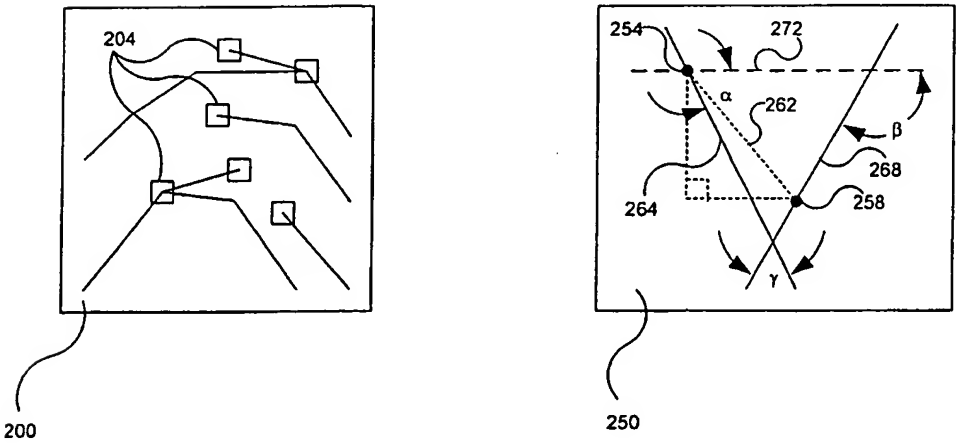

204
200
254
272
α
262
β
268
264
258
γ
250

FIG. 2

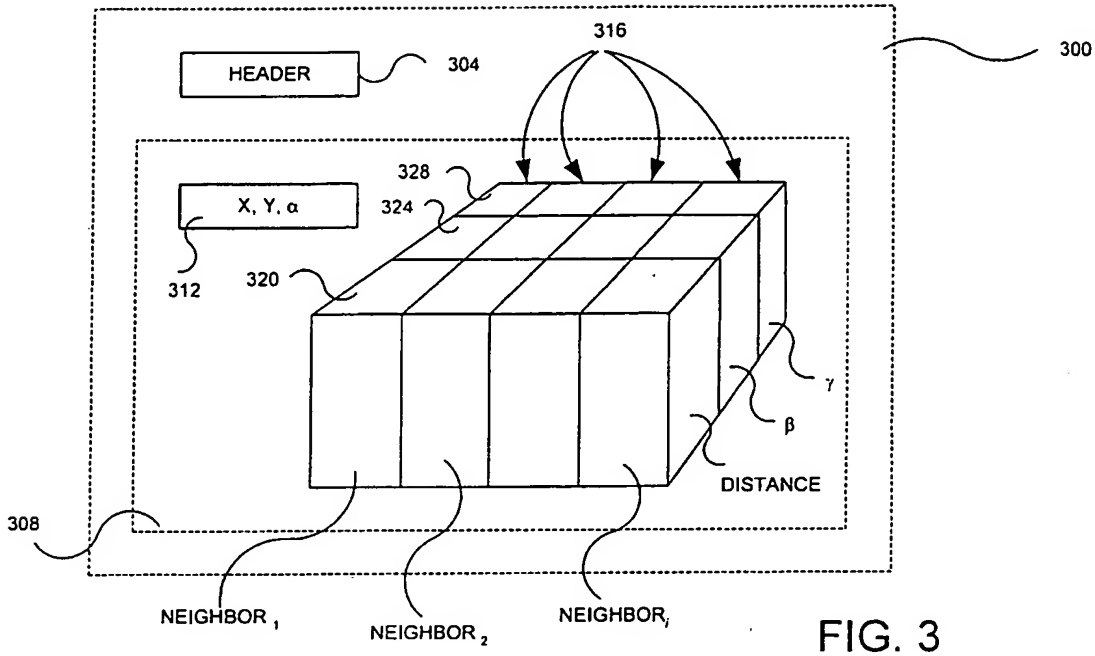

316
300
HEADER
304
328
X, Y, α
324
312
320
γ
β
DISTANCE
308
NEIGHBOR 1
NEIGHBOR 2
NEIGHBOR i

FIG. 3

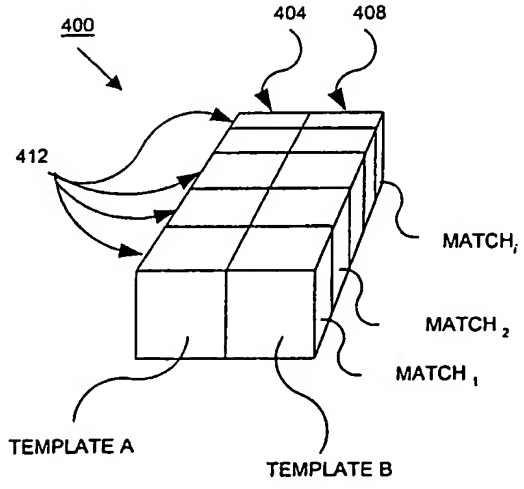

400
404
408
412
MATCH i
MATCH 2
MATCH 1
TEMPLATE A
TEMPLATE B

FIG. 4

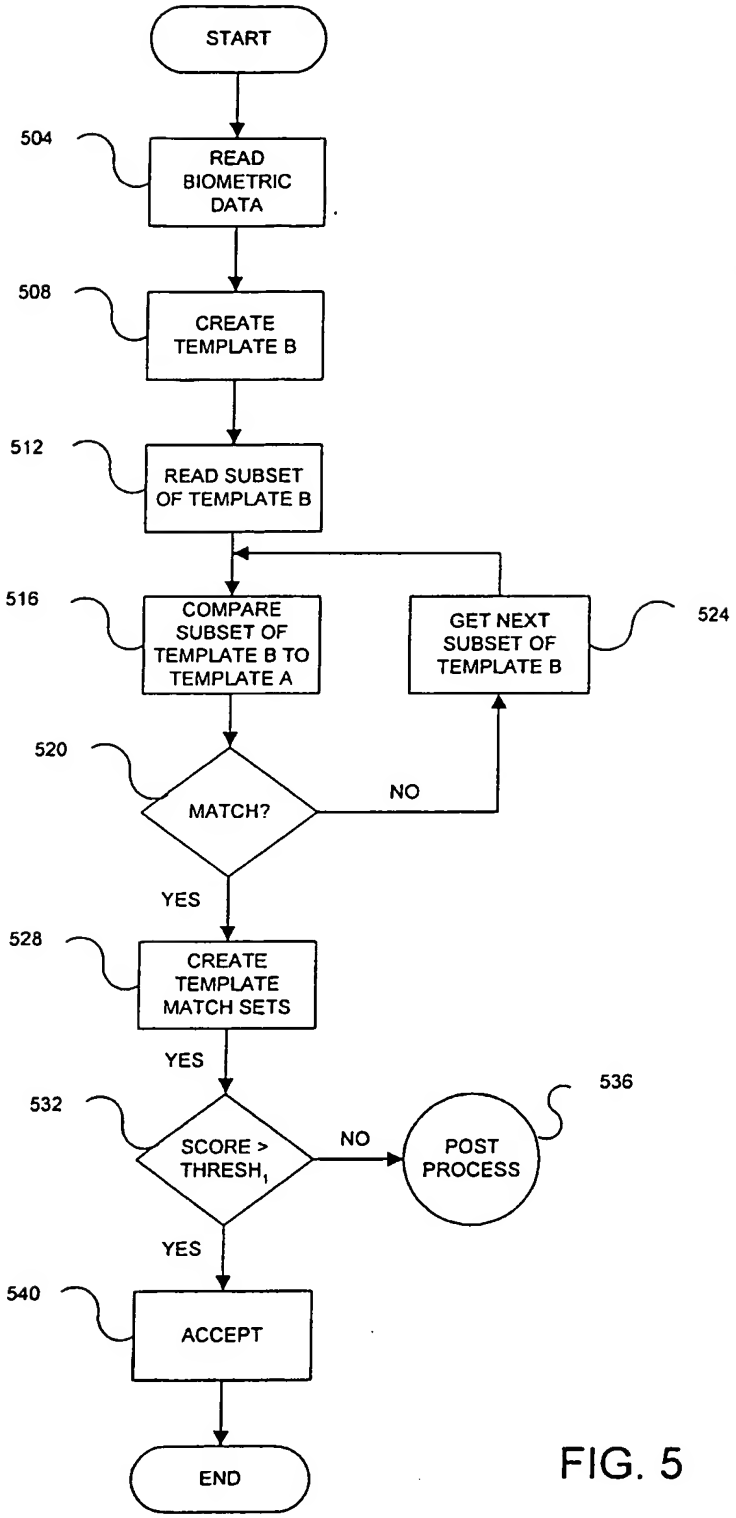

START
504
READ BIOMETRIC DATA
508
CREATE TEMPLATE B
512
READ SUBSET OF TEMPLATE B
516
COMPARE SUBSET OF TEMPLATE B TO TEMPLATE A
524
GET NEXT SUBSET OF TEMPLATE B
520
MATCH?
NO
YES
528
CREATE TEMPLATE MATCH SETS
YES
532
SCORE > THRESH1
NO
536
POST PROCESS
YES
540
ACCEPT
END

FIG. 5

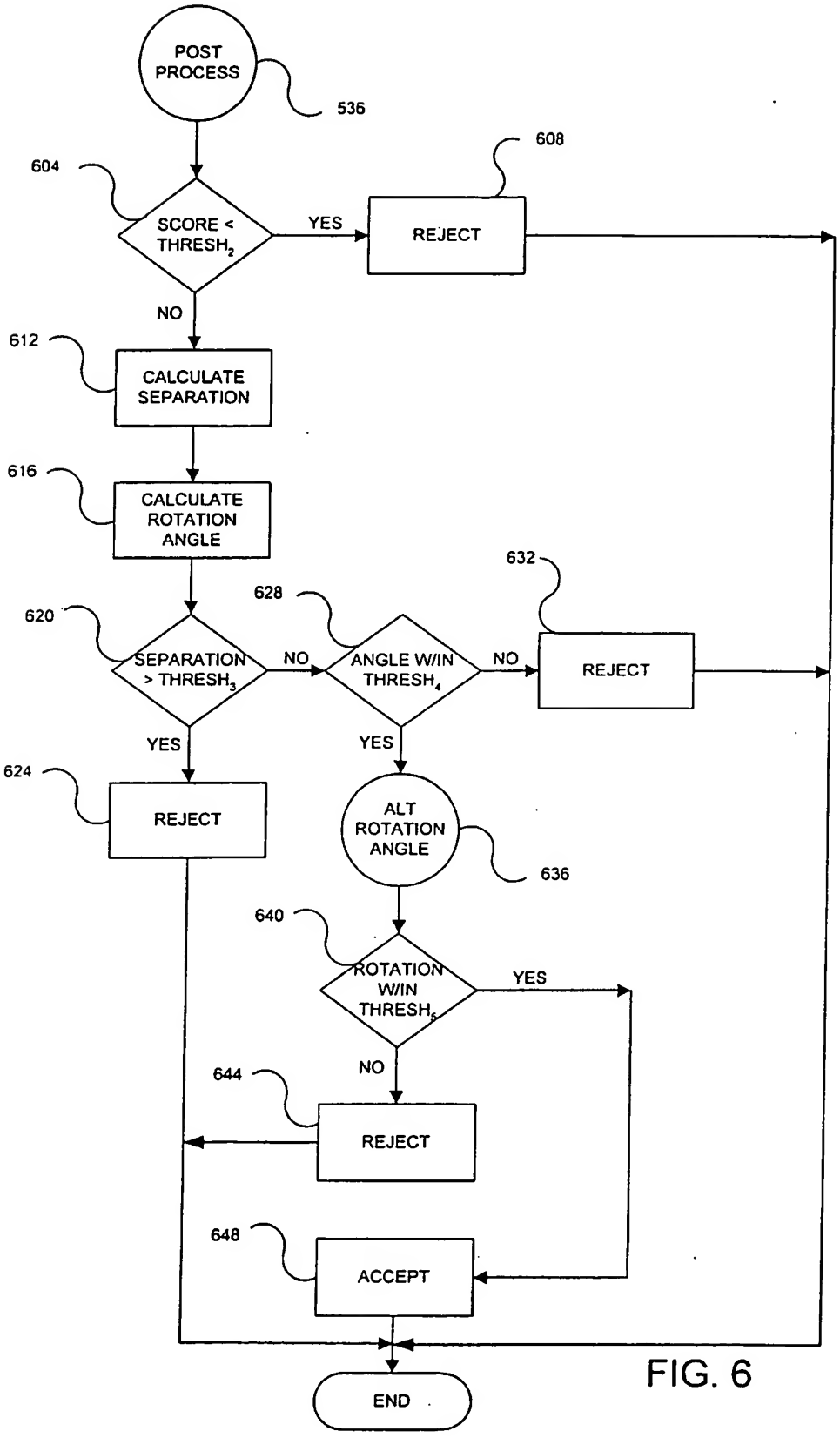

POST PROCESS
536
604
SCORE < THRESH2
YES
608
REJECT
NO
612
CALCULATE SEPARATION
616
CALCULATE ROTATION ANGLE
620
SEPARATION > THRESH3
NO
628
ANGLE W/IN THRESH4
NO
632
REJECT
YES
YES
624
REJECT
ALT ROTATION ANGLE
636
640
ROTATION W/IN THRESH5
YES
NO
644
REJECT
648
ACCEPT
END

FIG. 6

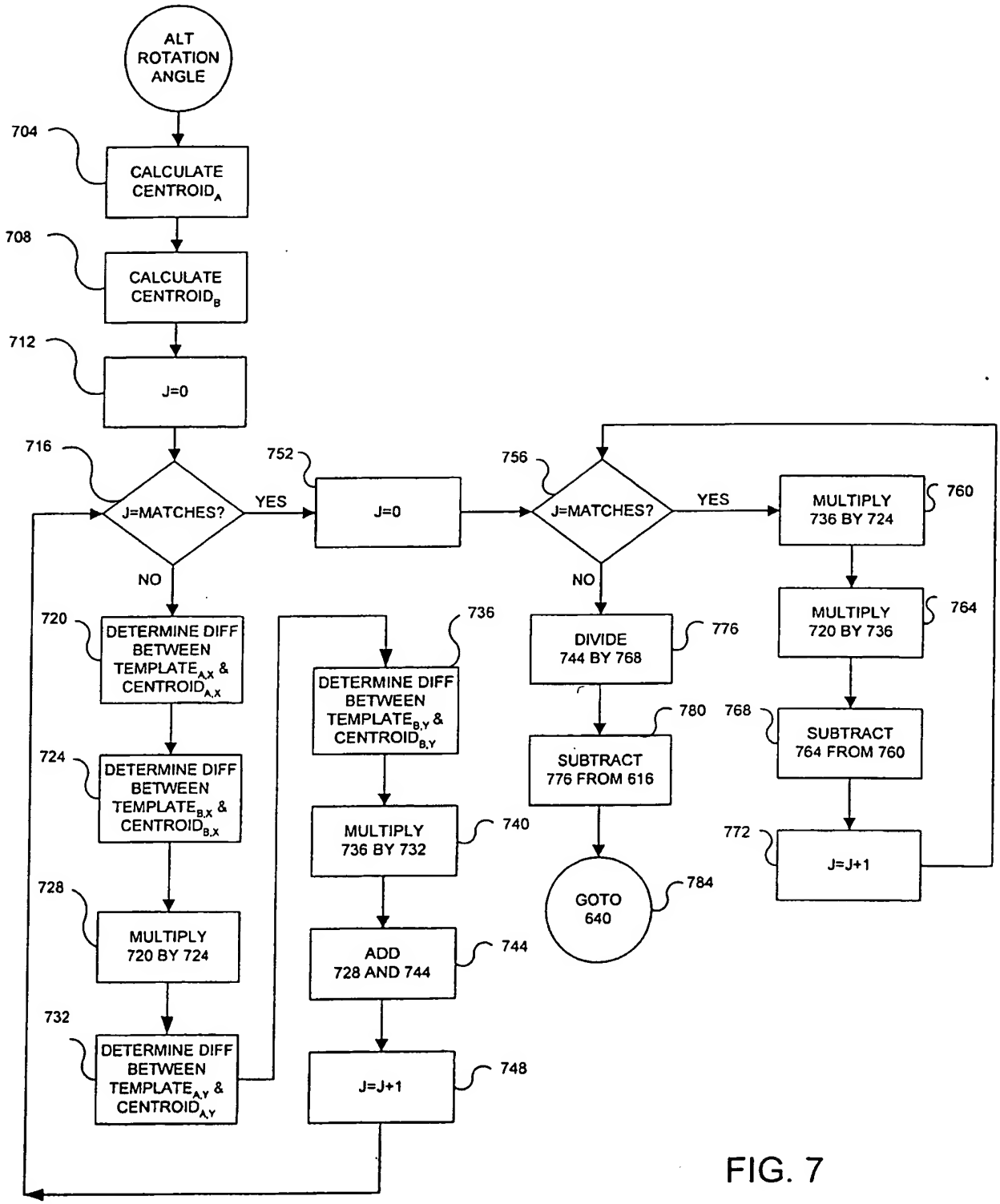

ALT ROTATION ANGLE
704
CALCULATE CENTROID$_A$
708
CALCULATE CENTROID$_B$
712
J=0
716
J=MATCHES?
YES
NO
752
J=0
720
DETERMINE DIFF BETWEEN TEMPLATE$_{A,X}$ & CENTROID$_{A,X}$
724
DETERMINE DIFF BETWEEN TEMPLATE$_{B,X}$ & CENTROID$_{B,X}$
728
MULTIPLY 720 BY 724
732
DETERMINE DIFF BETWEEN TEMPLATE$_{A,Y}$ & CENTROID$_{A,Y}$
736
DETERMINE DIFF BETWEEN TEMPLATE$_{B,Y}$ & CENTROID$_{B,Y}$
740
MULTIPLY 736 BY 732
744
ADD 728 AND 744
748
J=J+1
756
J=MATCHES?
YES
NO
760
MULTIPLY 736 BY 724
764
MULTIPLY 720 BY 736
768
SUBTRACT 764 FROM 760
772
J=J+1
776
DIVIDE 744 BY 768
780
SUBTRACT 776 FROM 616
784
GOTO 640

FIG. 7

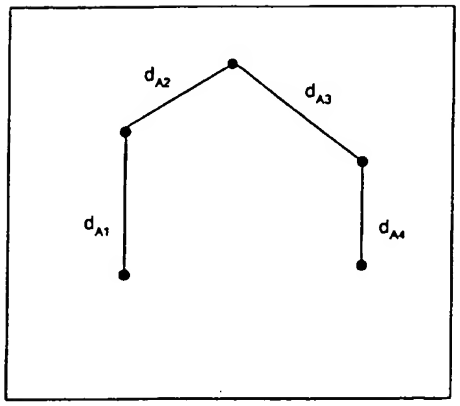


FIG. 8A

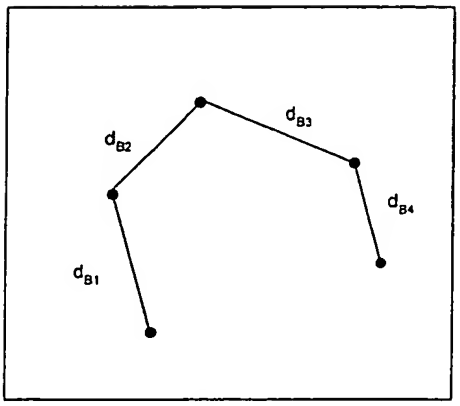


FIG. 8B

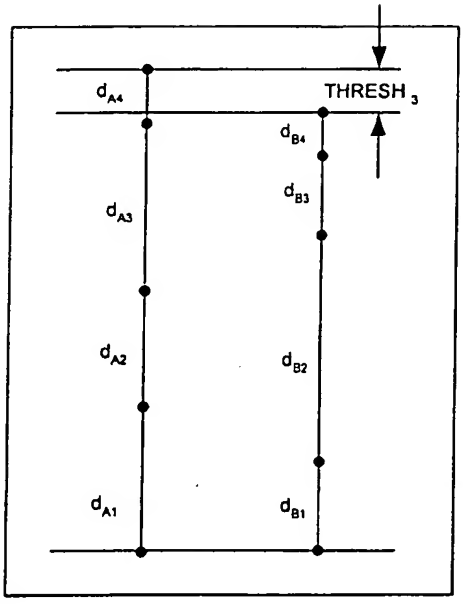


FIG. 8C

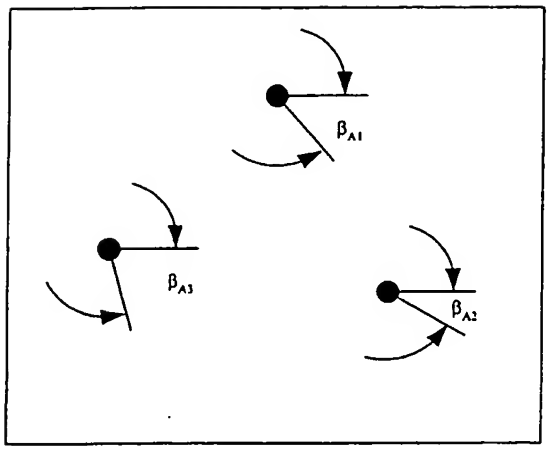


FIG. 9A

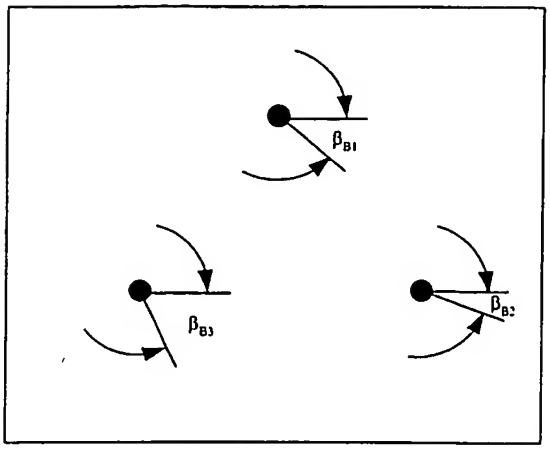


FIG. 9B

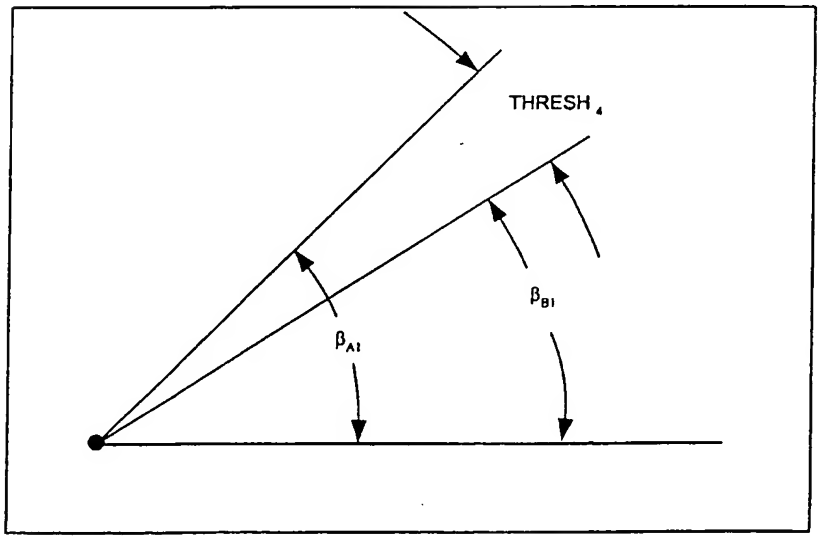


FIG. 9C

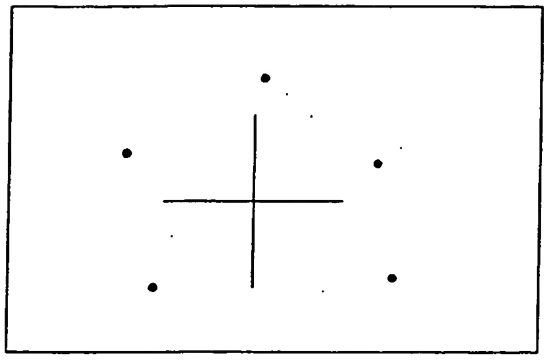


FIG. 10A

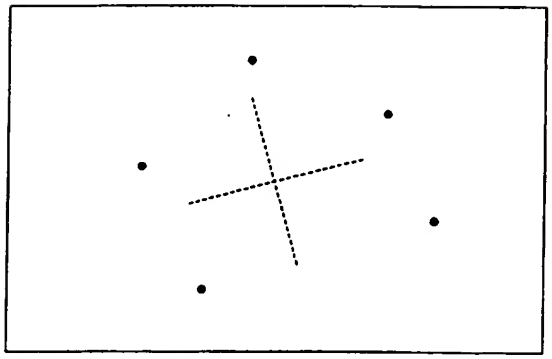


FIG. 10B

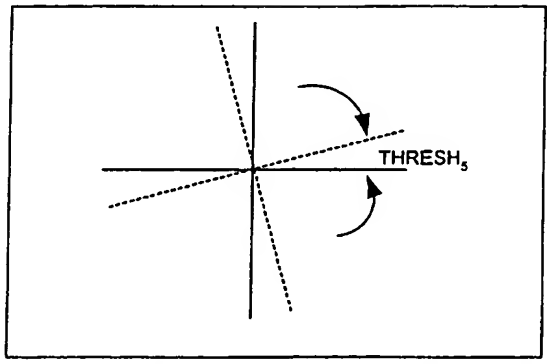


FIG. 10C